

# PATENT ABSTRACTS OF JAPAN

(11)Publication number : 10-240576  
 (43)Date of publication of application : 11.09.1998

(51)Int.CL.

G06F 11/28

G06F 9/06

(21)Application number : 09-045144

(71)Applicant : SONY CORP

(22)Date of filing : 28.02.1997

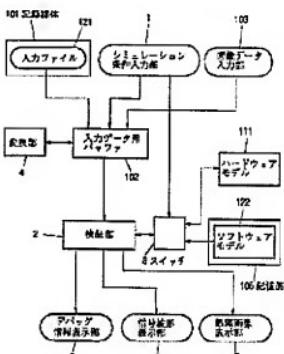
(72)Inventor : AOYAMA KOJI  
KONDO YOSHITO

## (54) VERIFICATION DEVICE AND METHOD

### (57)Abstract:

**PROBLEM TO BE SOLVED:** To verify the respective functions of a DSP(digital signal processor) in details in a short time.

**SOLUTION:** The data of verification conditions are inputted to a simulation condition input part 1 and an application program is supplied from a recording medium 101 to a buffer 102 for input data first. Then, the inputted data of the conditions and image data inputted to an image data input part 103 are supplied to the buffer 102 for the input data. Then, the buffer 102 for the input data outputs the application program described by an instruction set for the DSP to a verification part 2 along with the data and the verification part 2 performs verification corresponding to the verification conditions supplied from the buffer 102 for the input data. Then, the verification part 2 outputs debugging information, signal waveform information and the image data after a processing which are the results of the verification respectively to a debugging information display part 5, a signal waveform information display part 6 and a processed image display part 7.



(19) 日本国特許庁 (J P)

## (12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平10-240576

(13) 公開日 平成10年(1998) 9月11日

(51) Int.Cl.<sup>6</sup>  
 G 0 6 F 11/28 3 4 0  
 9/06 5 4 0

F I  
 G 0 6 F 11/28 3 4 0 A  
 9/06 5 4 0 U

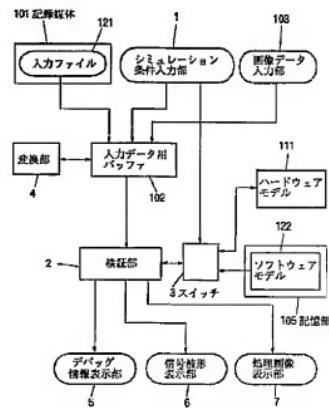
(21) 出願番号 特願平9-45144  
 (22) 出願日 平成9年(1997) 2月28日

(71) 出願人 0000021S5  
 ソニー株式会社  
 東京都品川区北品川6丁目7番35号  
 (72) 発明者 青山 幸治  
 東京都品川区北品川6丁目7番35号 ソニ  
 一株式会社内  
 (72) 発明者 近藤 芳人  
 東京都品川区北品川6丁目7番35号 ソニ  
 一株式会社内  
 (74) 代理人 弁理士 稲本 義雄

## (54) 【発明の名称】 検証装置および方法

## (57) 【要約】

【課題】 短時間で D S P の各機能を詳細に検証する。  
 【解決手段】 最初に、シミュレーション条件入力部 1 に検証条件のデータが入力されるとともに、記憶媒体 1 0 1 からアプリケーションプログラムが入力データ用バッファ 1 0 2 に供給される。そして、入力された条件のデータと、画像データ入力部 1 0 3 に入力された画像データが、入力データ用バッファ 1 0 2 に供給される。そして、入力データ用バッファ 1 0 2 は、これらのデータとともに、D S P 用の命令セットで記述されたアプリケーションプログラムを検証部 2 に当てる。検証部 2 は、入力データ用バッファ 1 0 2 により供給された検証条件に従って検証を行う。そして、検証部 2 は、検証の結果であるデバッグ情報、信号波形情報、および、処理後の画像データを、デバッグ情報表示部 5、信号波形情報表示部 6、および、処理画像表示部 7 にそれぞれ出力する。



検証装置

## 【特許請求の範囲】

【請求項1】 所定の処理回路のソフトウェアモデルまたはハードウェアモデルを利用して、前記処理回路のアプリケーションプログラムを検証する検証装置において、

前記ソフトウェアモデルおよび前記ハードウェアモデルを使用して、前記アプリケーションプログラムを検証する検証手段と、

前記検証手段により検証における検証項目を設定する設定手段とを備えることを特徴とする検証装置。

【請求項2】 前記ハードウェアモデルを構成し、前記処理回路に応じて処理を行う処理手段と、前記ソフトウェアモデルを記憶する記憶手段と、前記処理手段のハードウェアモデルおよび前記記憶手段のソフトウェアモデルのうち、前記検証において使用する1つまたは2つのモデルを設定する第2の設定手段とをさらに備えることを特徴とする請求項1に記載の検証装置。

【請求項3】 前記所定の処理回路は、デジタルシグナルプロセッサであることを特徴とする請求項1に記載の検証装置。

【請求項4】 前記アプリケーションプログラムは、画像処理用のプログラムであることを特徴とする請求項1に記載の検証装置。

【請求項5】 所定の処理回路のソフトウェアモデルまたはハードウェアモデルを利用して、前記処理回路のアプリケーションプログラムを検証する検証方法において、

検証における検証項目を設定するステップと、前記ソフトウェアモデルおよび前記ハードウェアモデルを使用して、前記アプリケーションプログラムを前記検証項目に関して検証するステップとを備えることを特徴とする検証方法。

## 【発明の詳細な説明】

## 【0001】

【発明の属する技術分野】 本発明は、検証装置および方法に問い合わせ、特に、検証における検証項目を設定し、ソフトウェアモデルおよびハードウェアモデルを使用して、アプリケーションプログラムを、その構造上に關して検証する検証装置および方法に関する。

## 【0002】

【従来の技術】 所定の画像処理を実行する回路を実現する場合、その処理専用のASIC(Application Specific Integrated Circuit)が設計されることが多い。

【0003】一方、最近の半導体技術の進歩に伴い、処理を記述したソフトウェアを利用してプログラマブル回路を実現するDSP(Digital Signal Processor)の処理速度が速くなりつつあり、実時間で画像処理が可能である速度を有するDSPが開発されている。

【0004】 DSPにより処理されるアプリケーション

プログラムの開発においては、プログラムの検証を行うことが必要であり、検証を行う場合においては、DSPの機能をソフトウェア化したプログラムであるソフトウェアモデル、あるいは、DSPの機能を有する回路であるハードウェアモデルが利用される。

【0005】 図7は、DSPのソフトウェアモデルを利用した検証装置の構成例を示している。記録媒体101には、アプリケーションプログラムを有する入力ファイル121が記録されており、そのプログラムは、入力データ用バッファ102に出力される。

【0006】 画像データ入力部103は、所定の装置(図示せず)より供給された所定の画像データを入力データ用バッファ102に出力する。

【0007】 入力データ用バッファ102は、供給されたアプリケーションプログラムと画像データを一時的に保持する。

【0008】一方、記憶部104には、DSPの機能を有するプログラムであるソフトウェアモデル122が記憶されており、このソフトウェアモデル122が検証部104に出力される。

【0009】このとき、入力データ用バッファ102から、アプリケーションプログラムと画像データも、検証部104に出力される。

【0010】 検証部104は、いわゆるマイクロコンピュータとして構成されており、供給されたアプリケーションプログラムとソフトウェアモデルに従って、画像データに対してDSPにより行われる処理の検証結果を表示部106に表示させる。

【0011】このとき、デバッグ情報、信頼波形情報、および、処理した画像情報のいずれか1つが、検証結果として出力されるように予め設定されており、設定された情報だけが、表示部106に表示される。

【0012】 図8は、DSPのハードウェアモデルを利用した検証装置の構成例を示している。記録媒体101、入力データ用バッファ102、および、画像データ入力部103は、図7のものと同様であり、アプリケーションプログラムおよび画像データが、入力データ用バッファ102を介して検証部112に供給される。

【0013】 ハードウェアウェアモデル111は、実際に使用するDSPの回路(LSIチップ)を内蔵している。

【0014】 検証部112は、入力データ用バッファ102より供給されたアプリケーションプログラムに従って処理を行い、その処理のうち、DSPにより行われる処理をハードウェアモデル111に行わせることにより検証を行う。

【0015】 この場合の検証結果は、実際のハードウェア(DSP)を使用しているので、処理した画像情報だけであり、この画像情報は、処理画像表示部113(例えばCRT)に表示される。

【0016】このようにして、ハードウェアモデルまたはソフトウェアモデルを利用して、DSP用のアプリケーションプログラムの検証が行われている。

#### 【0017】

【明細が解決しようとする課題】しかしながら、ソフトウェアモデルを利用して検証を行う場合、DSPの各機能に対する詳細な情報を獲得することができるが、検証に要する処理時間が非常に長い。一方、ハードウェアモデルを利用して検証を行う場合、実時間で処理が行われるので検証に要する時間は短いが、DSPの各機能に対する詳細な情報を獲得することができ難である。従って、短時間でDSPの各機能を詳細に検証することが困難であるという問題を有している。

【0018】本発明は、そのような状況に鑑みてなされたもので、検証における検証項目を設定し、ソフトウェアモデルおよびハードウェアモデルを使用して、アプリケーションプログラムをその検証項目に關して検証するようにして、短時間でDSPの各機能を詳細に検証することができるようにするものである。

#### 【0019】

【課題を解決するための手順】請求項1に記載の検証装置は、ソフトウェアモデルおよびハードウェアモデルを使用して、アプリケーションプログラムを検証する検証手段と、検証手段により検証における検証項目を設定する設定手段とを備えることを特徴とする。

【0020】請求項1に記載の検証方法は、検証における検証項目を設定するステップと、ソフトウェアモデルおよびハードウェアモデルを使用して、アプリケーションプログラムを検証手段に關して検証するステップとを備えることを特徴とする。

【0021】請求項1に記載の検証装置においては、設定手段は、検証手段により検証における検証項目を設定し、検証手段は、ソフトウェアモデルおよびハードウェアモデルを使用して、アプリケーションプログラムを検証手段に關して検証する。

【0022】請求項5に記載の検証方法においては、検証における検証項目を設定し、ソフトウェアモデルおよびハードウェアモデルを使用して、アプリケーションプログラムを検証手段に關して検証する。

【0023】  
【明細の実施の形態】図1は、本発明の検証装置の一実施の形態の構成を示している。この検証装置における記録媒体101、入力データ用バッファ102、画像データ入力部103、記憶部105(記憶手段)、および、ハードウェアモデル111(処理手段)は、図7または図8のものと同じであるのでその説明を省略する。

【0024】シミュレーション条件入力部1(設定手段)は、例えばSD(Standard Definition)画像、HD(High Definition)画像などの、入力する画像データの種類、検証時にデバッグ情報として出力させるDS

Pの機能、検証時に波形情報がに出力されるDSPの所定の部位、検証に利用するモデルの種類などの検証の条件を入力され、モデルの種類に対応する信号をスイッチ3(第2の設定手段)に出力するとともに、これらの検証の条件を入力データ用バッファ102に出力するようになされている。

【0025】変換部4は、記録媒体101から入力データ用バッファ102に供給されたアプリケーションプログラムが、例えばアンブリ語言やC言語などで記述されている場合、即ち、検証するDSP用の命令セット(機械語)で記述されていない場合、そのアプリケーションプログラムを検証するDSP用の命令セットに変換するようになされている。

【0026】スイッチ3は、シミュレーション条件入力部1より供給された信号に対応して、ソフトウェアモデル102を記憶している配電部105またはハードウェアモデル111を検証部2(検証手段)に接続させるようになされている。

【0027】検証部2は、入力データ用バッファ102より供給された検証条件で、アプリケーションプログラムに従って処理を行い、検証条件により指定されている信号のうち、デバッグ情報をデバッグ情報表示部6に出力し、波形情報を信号波形表示部6に出力し、処理した画像データを処理画像表示部7に出力するようになされている。

【0028】デバッグ情報表示部5は、例えばCRTを有しており、供給されたデバッグ情報をそのCRTに表示させるようになされている。

【0029】信号波形表示部6は、例えばロジックアナライザであり、供給された波形情報を対応する信号波形を例えばCRTに表示せるようになされている。

【0030】処理画像表示部7は、例えばCRTを有しており、供給された画像データに対応する画像をそのCRTに表示させるようになされている。

【0031】次に、図2のフローチャートを参照して、図1の検証装置の動作について説明する。

【0032】最初に、ステップS1において、シミュレーション条件入力部1に、例えば図3に示すような条件のデータが入力される。図3に示す条件のデータにおいては、SD画像が、入力する画像データの種類として設定され、DSPの加算器のレジスタ値、DSPの除算器のレジスタ値、および、DSPのプログラムカウンタの値が、デバッグ情報として設定されている。さらに、図3のデータにおいては、DSPの演算器の1/0ピンの波形が、波形情報として設定され、ソフトウェアモデルとハードウェアモデルの両方が、検証において使用されるモデルとして設定されている。

【0033】なお、この条件のデータを入力する場合、例えばキーボードなどを利用して入力するようにしてもよいし、図3に示すようにテキストデータとして所定の

記録媒体にファイルとして予め保存しておき、そのファイルを入力するようにしてもよい。

【0034】次に、ステップS2において、記録媒体101からアプリケーションプログラムが入力データ川バッファ102に供給される。アプリケーションプログラムが、検証するDSP用の命令セットで記述されていない場合、変換部4は、そのアプリケーションプログラムを、DSP用の命令セットで記述されたものに変換する。

【0035】例えば、図4に示すように画像サイズを横幅それぞれ2倍にする処理を行う場合、新たに挿入される画素の値L1乃至L5は、次式に従って算出される。

$$L_1 = (P_1 + P_2) / 2$$

$$L_2 = (P_3 + P_4) / 2$$

$$L_3 = (P_1 + P_3) / 2$$

$$L_4 = (P_2 + P_4) / 2$$

$$L_5 = (L_1 + L_2) / 2$$

ここで、P1乃至P4は、元の画像のうちの図4に示す4つの画素の画素値を表し、L1乃至L5は、拡大後の画像のうちの、図4に示す元の画像のうちの4つの画素の画素値より算出される画素値(即ち、新たに挿入される画素の値)を表している。

【0036】そして、この処理に対応するアプリケーションプログラムは、図5に示すように、例えばDSP用のアセンブリ言語で記述される。なお、図5において、行頭の「;」は、その行がコメント行であることを表し、命令「add r1, x1, x2」は、レジスタまたは変数x1の値とレジスタまたは変数x2の値の和を計算し、その計算結果をレジスタまたは変数r1に保存することを表し、命令「div r1, x1, x2」は、レジスタまたは変数x1の値を2で除算し、その計算結果をレジスタまたは変数r1に保存することを表している。

【0037】このように、アプリケーションプログラムがアセンブリ言語で記述されている場合、変換部4は、このアセンブリ言語で記述されたアプリケーションプログラムをDSP用の命令セット(構成版)で記述されたものに変換する。

【0038】例えば、DSPの32ビットの命令セットにおいて、上位6ビット(第32ビット乃至第27ビット)がオペコードを表し、次の上位6ビット(第26ビット乃至第21ビット)が演算結果を代入するレジスタのアドレスを表し、第20ビット乃至第15ビットの6ビットと、第14ビット乃至第9ビットの6ビットが、演算する値を格納しているレジスタのアドレスをそれぞれ表し、第8ビット乃至第1ビットが、演算する値の単位を表すものとする。その場合、命令「add」のオペコードを「000001」とし、命令「div」のオペコードを「010010」とし、「r」で始まるレジスタのタイプを「00」とし、「P」で始まるレジスタのタイプを「01」とし、「L」で始まるレジスタ

のタイプを「10」とすると、変換部4は、図5に示すアセンブリ言語で記述されたプログラムを、図6に示す機械語の命令セット(16進数)に変換する。なお、図6の「キ」以降の文字列は、コメントであり、特に意味はない。

【0039】なお、DSPの命令セットの語長(ビット長)は、32ビットに限定されるものではなく、16ビットや他のビット長でもよい。

【0040】また、このとき、ステップS1において入力された条件のデータと、画像データ入力部103に入力された画像データが、入力データ用バッファ102に供給される。なお、画像データは、検証条件において設定されているれば、静止画像のデータおよび動画像のデータのいずれでもよい。

【0041】そして、入力データ用バッファ102は、これらのデータとともに、DSP用の命令セットで記述されたアプリケーションプログラムを検証部2に出力する。

【0042】次に、ステップS3において、検証部2は、入力データ川バッファ102より供給された検証条件に従って検証を行う。このとき、検証条件において設定されているDSPのモデルがソフトウェアモデルである場合、検証部2は、記憶部105が記憶しているソフトウェアモデル122を使用して検証を行う。

【0043】なお、ソフトウェアモデル122として、C言語で記述されたものの他、Verilog HDL(Hardwared Description Language)やVHDL(VHSIC HDL)などのハードウェア記述言語で記述されたものを使用することができる。なお、ソフトウェアモデル122がC言語で記述されている場合、検証部2は、そのソフトウェアモデルを、Cコンパイラを利用して機械語に変換した後、検証を行う。また、ソフトウェアモデル122がハードウェア記述言語で記述されている場合、検証部2は、そのハードウェア記述言語に対応するシミュレータを利用して検証を行う。

【0044】一方、検証条件において設定されているDSPのモデルがハードウェアモデルである場合、検証部2は、ハードウェアモデル111にアプリケーションプログラムを実行させる。

【0045】また、検証条件において設定されているDSPのモデルがソフトウェアモデルとハードウェアモデルの両方である場合、検証部2は、予め設定されている部分の処理をソフトウェアモデル122に従って実行し、その他の処理をハードウェアモデル111に実行させる。

【0046】なお、検証条件において設定されているDSPのモデルがソフトウェアモデルとハードウェアモデルの両方である場合、検証する処理のうち、ソフトウェアモデル122により検証される部分と、ハードウェアモデル111により検証される部分は、検証条件とし

て、シミュレーション条件入力部1に入力されるデータにおいて設定される。

【0047】例えば、図3に示すように検証条件が設定され、DSPに図6に示す処理を行わせる場合、デバッグ情報であるDSP内の加算器のレジスタ値、除算器のレジスタ値、および、プログラムカウントの値、並びに、信号波形情報であるDSPの演算器の1/Oピンの信号波形は、ソフトウェアモデル122を利用して算出し、処理後の画像データは、ハードウェアモデル111を利用して算出する。

【0048】そして、ステップS4において、検証部2は、検証の結果であるデバッグ情報、信号波形情報、および、処理後の画像データを、デバッグ情報表示部5、信号波形情報表示部6、および、処理画像表示部7にそれぞれ出力する。

【0049】なお、このとき、同時に、デバッグ情報、信号波形情報、および、処理後の画像データを、デバッグ情報表示部5、信号波形情報表示部6、および、処理画像表示部7にそれぞれ表示させることができる。

【0050】以上のようにして、ハードウェアモデルとソフトウェアモデルを利用して、シミュレーション条件入力部1に入力された検証条件に従ってアプリケーションプログラムの検証が行われる。

【0051】なお、上記実施の形態は、DSPのアプリケーションプログラムの検証を行っているが、ハードウェアモデルとソフトウェアモデルを変更することにより、他の処理回路の検証を行うことが可能である。

【0052】

【発明の効果】以上のごとく、請求項1に記載の検証装置および請求項5に記載の検証方法によれば、検証における検証項目を設定し、ソフトウェアモデルおよびハードウェアモデルを使用して、アプリケーションプログラムを検証項目に従って検証するので、短時間でDSPの各機能を詳細に検証することができる。

【図面の簡単な説明】

【図1】本発明の検証装置の一実施の形態の構成例を示すブロック図である。

【図2】図1の検証装置の動作について説明するフローチャートである。

【図3】検証条件の一例を示す図である。

【図4】画像の大規模の一例を示す図である。

【図5】アセンブリ言語で記述されたアプリケーションプログラムの一例を示す図である。

【図6】機械語に変換された図5のアプリケーションプログラムの一例を示す図である。

【図7】ソフトウェアモデルだけを利用した検証装置の一構成例を示すブロック図である。

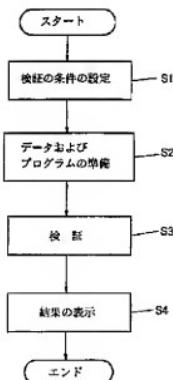
【図8】ハードウェアモデルだけを利用した検証装置の一構成例を示すブロック図である。

【符号の説明】

1 シミュレーション条件入力部, 2 検証部, 3

スイッチ, 4 要換部, 5 デバッグ情報表示部, 6 信号波形表示部, 7 処理画像表示部, 101 記録媒体, 102 入力データ用バッファ, 105 記憶部, 111 ハードウェアモデル, 122 ソフトウェアモデル

【図2】



【図3】

| 画像の種類:     |   |
|------------|---|
| デバッグ情報:    | 【DSP内の加算器のレジスタ値】<br>【DSP内の除算器のレジスタ値】<br>【DSP内のプログラムカウント値】 |
| 波形情報:      | 【DSP内の演算器の1/Oピンの全信号に對応する値】                                |
| DSPモデルの種類: | 【ハードウェアとソフトウェアの両方に對応する値】                                  |
| 検証条件の一例    |   |

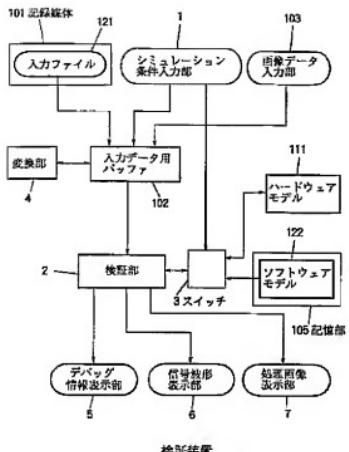
【図5】

```

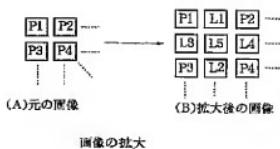
%L1=(P1+P2)/2
add r1, P1, P2;
div r1, L1, r1;
%L2=(P3+P4)/2
add r1, P3, P4;
div r1, L2, r1, 2;
%L3=(P1+P3)/2
add r1, P1, P3;
div r1, L3, r1, 2;
%L4=(P2+P4)/2
add r1, P2, P4;
div r1, L4, r1, 2;
%L5=(L1+L2)/2
add r1, L1, L2;
div r1, L5, r1, 2;

```

【図1】



【図4】



画像の拡大

【図6】

```

04145200 # add r1, P1, P2;
4A104002 # divi L1, r1, 2;

04145800 # add r1, P3, P4;
4A204002 # divi L2, r1, 2;

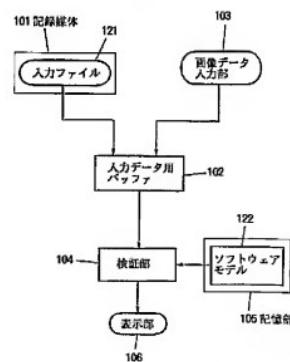
04140400 # add r1, P1, P2;
4A304002 # divi L3, r1, 2;

04149400 # add r1, P2, P4;
4A404002 # divi L4, r1, 2;

04185200 # add r1, L1, L2;
4A504002 # divi L5, r1, 2;

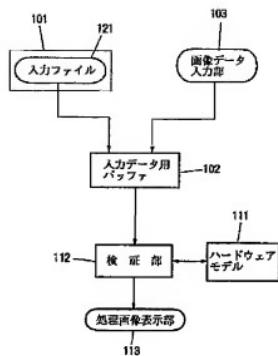
```

【図7】



ソフトウェアモデルだけを利用した検証装置

【図8】



ハードウェアモデルだけを利用した検証装置